

审核流程使用说明及注意事项

3 功能说明

3.1 数据集制作和电子表格制作

参考Wiki相关文档。

3.2 流程定义

3.2.1 流程定义的入口一是在定制管理的面板菜单中，入口二是在电子表格的右键菜单中。



3.2.2 流程定义界面包括流程基本信息定义和流程节点定义两个步骤：

说明如下：

- 电子表格id：
选择用来展示审核数据、回写数据的电子表格。
电子表格可以多选，意思是这些电子表格使用相同的审核流程定义。

一个电子表格只能绑定一个流程，新的绑定设置会覆盖旧的绑定设置。

- 汇总相关的机构参数：
如果数据审核中需要从下而上汇总数据，则需要选择一个和机构参数，而且此参数的结构需要和用户管理中的用户组结构相同，否则无法正确汇总。一般情况下，机构参数的参数类型为下拉树，而且真实值为用户组的组编号，显示值为用户组的name。通常情况下，机构参数还需要与当前用户关联，只显示当前用户所属机构及下级机构。配置如下图：

显示参数 ☒ 允许多选 ☐ 手工输入 ☐ 校验合法性 ☐ 不显示标题 ☒ 输出合并

MDX使用引导 ☒ 允许为空 (存储过程查询) ☐ 参数源: 系统设置

构建树结构: 父子结构 ☐ 自动勾连子孙 ☒ 动态加载子节点 ☐ 检测子节点 ☒ 自定义父子关系 orgid

备选值设置:
SQL 对象 静态列表 函数
select g.c_orgid as orgid, g.c_groupname as groupname
from t_group g, t_group pg where pg.c_orgid=? and g.c_pgrougid = pg.c_grougid

默认值设置:
SQL 对象 静态列表 函数
select c.c_orgid, c.c_groupname as c_groupname
from t_group where c.c_orgid=CurrentUserDefaultDepartmentID { }

节点设置:
SQL 对象 静态列表 函数
select c.c_orgid, c.c_groupname as c_groupname
from t_group where c.c_orgid=CurrentUserDefaultDepartmentID { }

机构参数的可选范围，是已选电子表格中含有的参数。如果电子表格是多选，则是这些电子表格中的公有参数
机构相关参数和流程相关参数不能同时为空。

- 流程的相关参数：
用于唯一标识每一份审核的数据。常见的数据审核相关参数是时间，例如年份参数、年月参数、季度参数等。而对于报销单、请假单等OA应用，则需要选择单号参数。

流程相关参数的可选范围，是已选电子表格中含有的参数。如果电子表格是多选，则是这些电子表格中的公有参数。

流程相关参数和机构相关参数不能同时为空。

- 校验规则所在的Sheet：
输入校验规则所在的Sheet页名称。如果不填则校验功能无法生效。
- 自启动配置

类似计划任务，用于自动启动流程。配置后，自动生成对应任务和计划。当计划任务运行时，对于机构参数，按第一个节点的用户迭代；对于其它参数，取默认值。

- 每个sheet页一个流程

通常情况下，每个电子表格的多个sheet页数据，在相同参数下视为同一份审核数据，在整个审核过程中只有一个审核状态。

而勾选了“每个sheet页一个流程”选项后，每个sheet页的数据被视为一份独立的审核数据，有单独的审核状态和流转过程。

- 流程节点名称：用于标识每一步流程的节点名称
- 操作者：当前流程允许的操作人。可以选择具体用户、用户组或者角色
- 报表操作：当报表流转到这个节点的时候，报表上允许的操作，与电子表格展示界面上的功能按钮对应。
- 流程操作：在每一个流程步骤中，【上报】、【审核】、【发布】这三个用于控制流程向下流转的功能按钮，必须选择一个，而且只能选择一个
- 可见人设置（仅第一个填报节点有此设置）：

发起人可见

- （1）填报人使用填报角色，发起人可见，执行后在待办中只能看到发起人自己的流程
- （2）填报人使用填报用户组，发起人可见，执行后在待办中只能看到发起人自己的流程

(3) 填报人使用填报用户，发起人可见，执行后在待办中只能看到发起人自己的流程
所有人可见

(4) 使用角色，所有人可见，执行后该角色的用户在待办可以看到全部的流程

(5) 使用用户组，所有人可见，执行后该组的用户在待办可以看到全部的流程

(6) 使用用户，所有人可见，执行后设置的用户在待办可以看到全部的流程

同机构可见

(7) 使用角色，同机构可见，执行后该角色的用户在待办可以看到全部的流程

(8) 使用用户组，同机构可见，执行后该组的用户在待办可以看到全部的流程

(9) 使用用户，同机构可见，执行后和发起人同组的用户在待办可以看到全部的流程

注意：流程启动后修改可见人设置不生效，可见人必须在流程配置时设置才能保证有效。

3.2.3 电子表格的右键菜单上，除了【新建流程】，还有【绑定流程】和【取消绑定】功能。

- 绑定流程：为当前电子表格选择一个已有的流程定义。如果流程定义中已经选择参数，而且当前电子表格也有相同的参数，则绑定成功；如果流程定义中的参数在当前电子表格中不存在，则提示用户参数不同，绑定失败。
- 取消绑定：取消当前电子表格的绑定流程

3.3 流程启动

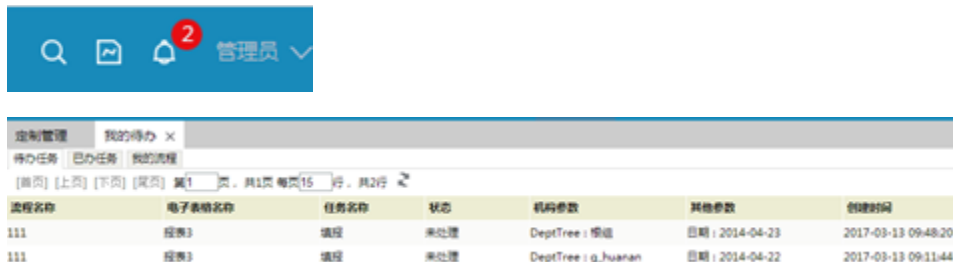
流程定义好以后，要启动流程开始流转，有三种操作方式。

- 方式1：在主界面右上角，点击【待办】菜单，在显示的界面中，点击【待办流程】选项卡，双击要启动的流程，跳转打开流程对应的电子表格，选择相应参数后，点击工具栏上的【保存】按钮，保存回写内容的同时启动流程。
- 方式2：和方式1类似，直接打开已经绑定流程的电子表格，选择相应参数后，点击工具栏上的【保存】按钮，保存回写内容的同时启动流程
- 方式3：在流程定义中配置了自启动相关信息，则由定时任务触发启动。

对于同一个报表，在相同的参数情况下，只能有一份正在流转的审核流程。因此，当上面三种方式之一启动之后，其它方式无法启动相同参数对应的流程。

3.4 流程待办

启动了流程之后，流程开始流转。在主界面右上角，点击【待办】菜单，在显示的界面中，点击【待办】选项卡，可以看到当前用户的待办任务。双击待办任务，打开对应的电子表格，并传递待办任务对应的参数到电子表格中。



切换到【已办任务】选项卡，可以查看当前用户已经处理过的流程。

切换到【我的流程】选项卡，可以查看所有当前用户可以发起的流程（即流程定义中第一个节点包含当前用户）。

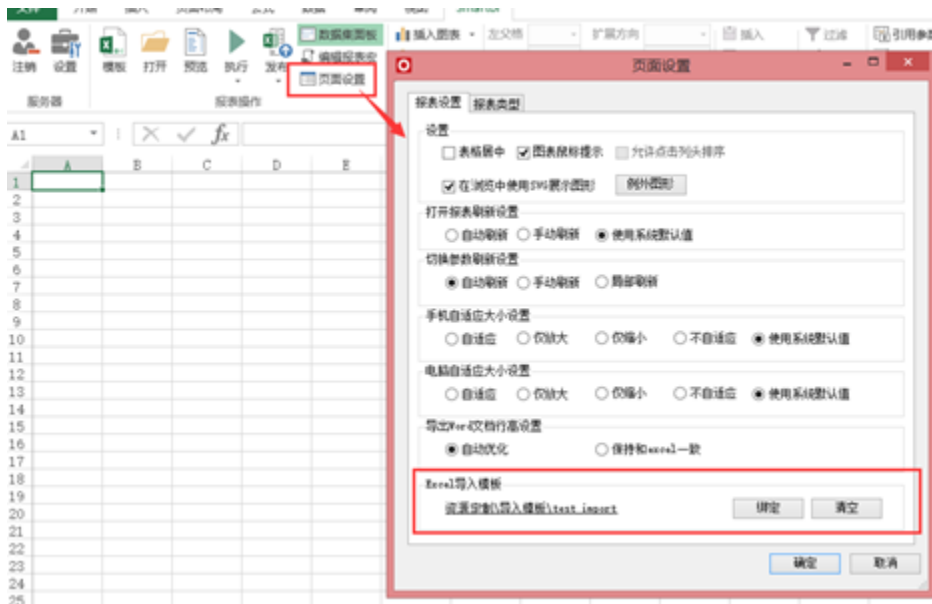
3.5 流程流转操作

流程的流转是通过电子表格展示时工具栏上的按钮来控制的。



1) 导入Excel

要使用【导入Excel】功能，需要先制作一份Excel导入模版，并在Excel中将该模版与电子表格关联。



绑定了导入模版之后，打开电子表格，界面上才有相应的按钮。点击按钮后，跳转到Excel模版的执行界面，同时传递当前所有参数的值，赋给Excel模版上的绑定参数。

2) 保存

【保存】按钮用于将当前数据按回写规则回写到数据库表中。同时，对于当前参数对应的审核数据还没有启动流程的情况，启动审核流程。流程启动后，当前流程状态对应的操作按钮会处于可用状态。

3) 校验

要使用【校验】按钮对数据进行校验，需要在制作电子表格的时候，用单独一个Sheet页的A列编写校验规则，并在流程定义的时候指定这个sheet页的名称。校验规则使用Excel的布尔表达式，例如“Sheet1!A2=Sheet1!B2+Sheet1!C2”，支持跨Sheet页取数，但不支持跨电子表格取数。

	A
1	
2	Sheet1!A2=Sheet1!B2+Sheet1!C2
3	Sheet1!A3=Sheet1!B3-Sheet1!C3
4	Sheet1!A4=Sheet1!A2+Sheet1!A3
5	Sheet1!A3=Sheet3!B2
6	

使用【校验】按钮后，会将当前Sheet页相关的校验不通过的单元格，在界面上用红色标识，并弹出提示。



4) 汇总

如果流程中含有【汇总】节点，且流转到该节点后，【汇总】按钮用。点击【汇总】，弹出窗口显示下级机构的上一个节点的审核情况。如果所有下级机构上个节点都已经完成，则可以点击【确定】按钮。确定之后，本节点的【提交】、【审核】、【发布】这几个流程相关按钮才可以使用。

5) 提交

执行【提交】操作，流程流转到下一节点

6) 审核

执行【审核】操作，流程流转到下一节点

7) 回退到开始

执行【回退到开始】操作，流程流转退回到第一个节点

8) 回退到上一步

执行【回退到上一步】操作，流程流转退回到上一节点

9) 发布

执行【发布】操作，流程流转到下一节点。实际情况中，【发布】往往是最后一个节点。

10) 填写备注。

弹出窗口填写备注信息

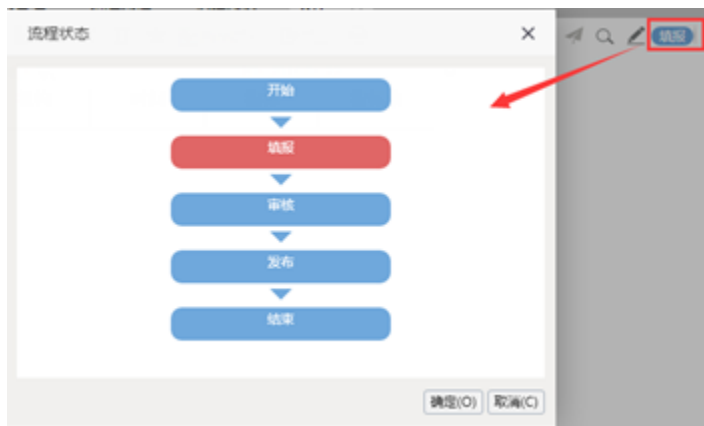
11) 查看流程日志

点击【查看流程日志】按钮，显示每一步流程操作的操作步骤和备注信息



12) 查看流程定义

电子表格的工具类上，审核功能操作按钮后面，显示当前参数标识的审核数据对应的流程状态，点击流程状态，显示整个流程步骤，并用红色标识当前所处的步骤

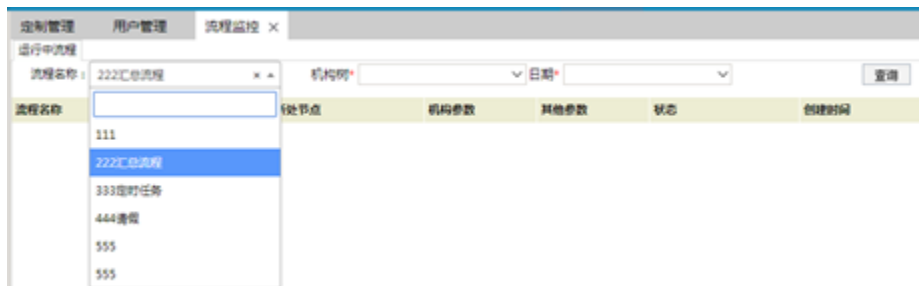


3.6 流程监控

要使用流程监控功能，必须先拥有【流程监控】的操作权限，如下图。



拥有流程权限后，在主界面右上角的下拉菜单中，可以看见【流程监控】菜单。点击菜单，进入流程监控主界面。在流程监控界面中，通过查询选择要监控的流程，展示流程对应的参数。不选择参数，展示所有的和当前流程相关的流程实例；选择参数，则查询展示与参数值相关的流程实例。



3.7 流程上的宏

宏事件往往用于帮助用户对数据进行处理。例如有些用户执行汇总的时候，需要执行复杂的存储过程进行数据汇总；或者在发布的时候，将数据从一个库表复制到另外一个库表；或者用户需要将当前流程的审核状态同步到业务库中，这些需求都可以通过宏来实现。

在流程定义上，右键编辑宏，可以进入到流程宏的编辑界面。流程只支持客户端宏，不支持服务器端宏。流程宏基本上与流程操作按钮对应，且按执行时间，分为执行按钮操作之前和执行按钮操作之后触发。流程支持的宏包括：

- beforeCommit: 执行提交之前
- afterCommit: 执行提交之后
- beforeVerify: 执行校验之前
- afterVerify: 执行校验之后
- beforeSummary: 执行汇总之前
- afterSummary: 执行汇总之后
- beforeAudit: 执行审核之前
- afterAudit: 执行审核之后
- beforeBackToPre: 执行回退上一步之前
- afterBackToPre: 执行回退上一步之后

- beforeBackToStart:执行回退到开始之前
- afterBackToStart:执行回到到开始之后
- beforeRelease:执行发布之前
- afterRelease:执行发布之后

特别的，在before类型的事件中，可以针对操作做一些条件判断，例如对业务库进行检查，当不满足条件的时候，可以通过抛出异常（throw “error”）的方法，阻止按钮操作的执行。

3.8 权限控制

要新建流程定义，需要【流程定义】操作权限，位于【定制管理】-》【新建分析】目录下。



另外，流程定义属于资源，本身受资源权限控制。

3.9 其它功能

流程定义支持资源导入、导出操作，支持在资源上复制、拖动换目录，支持影响性分析和血统分析。具体操作参考Wiki相关文档

4 特殊场景

4.1 业务库中查询已经发布的数据

很多情况下，除了审核流程相关的用户可以看到未经过审核的数据，其它用户是不允许查询这些未审核数据的。而在业务库中，这两种数据往往混合在一起存放。这个时候需要在查询中加上对流程状态的判断。可是由于流程状态是存在于知识库中的，跨库的查询是无法支持的。

遇到这种情况，可以由实施人员，在发布事件中编写宏代码，使用DataSourceService的executeUpdate方法，执行sql语句或者执行存储过程，将流程状态同步到业务库中。

```
var util = jsloader.resolve("freequery.common.util");

var ret = util.remoteInvoke("DataSourceService", "executeUpdate", ["DS. 数据源名称", "update XXXX"]);
```

4.2 发布时将数据写入另一个库

有些时候，用户要求审核前的数据和审核后的数据不能同库。这和4.1的处理类似，可以由实施人员，在发布事件中编写宏代码，使用DataSourceService的executeUpdate方法，执行sql语句或者执行存储过程，将审核后的数据同步到业务库中。

4.3 汇总时执行运算逻辑

系统中的【汇总】操作，只是做了一个状态标志的处理，并没真正的对业务数据做汇总处理。业务数据的汇总处理逻辑，往往逻辑复杂且不尽相同。

因此，可以由实施人员，在流程的汇总事件中编写宏代码，使用DataSourceService的executeUpdate方法，执行sql语句或者执行存储过程，完成响应的汇总处理。示例代码如下：

```
var p1 = spreadsheetReport.paramPanelObj.findParamByName('DeptTree');
var p2 = spreadsheetReport.paramPanelObj.findParamByName('');
var util = jsloader.resolve("freequery.common.util");
//
var sql = "delete from factyear where dateid='" + p2.value + "' and deptid='" + p1.value + "'";
var ret = util.remoteInvoke("DataSourceService", "executeUpdate", ["DS.", sql]);
if (ret && ret.succeeded) {
    //
    var sql = "insert into factyear (deptid,dateid,metricid,val) " +
        "select '" + p1.value + "'," + p2.value + "',metricid,sum(val) " +
        " from factyear where dateid='" + p2.value + "' and deptid<>'"+ p1.value +
        "' group by metricid ";

    var ret = util.remoteInvoke("DataSourceService", "executeUpdate", ["DS.", sql]);
    if (ret && ret.succeeded) {
        spreadsheetReport.doRefresh();
        return true;
    }
}
```